UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/087,950	03/05/2002	Moon Soo Lee	123056-05004487	5369

			EXAMINER
43569	7590	11/15/2006	ALHIJA, SAIF A

MAYER, BROWN, ROWE & MAW LLP
1909 K STREET, N.W.
WASHINGTON, DC 20006

ART UNIT	PAPER NUMBER
2128	

DATE MAILED: 11/15/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

PTO-90C (Rev. 10/03)

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on <u>22 September 2006</u>.

2a)☐ This action is **FINAL**.   2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) <u>1-6 and 8-15</u> is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) <u>1-6, and 8-15</u> is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on <u>05 March 2002</u> is/are: a)☒ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☒ All   b)☐ Some * c)☐ None of:

      1.☒ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____.

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☒ Notice of References Cited (PTO-892)

2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3)☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.

4)☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.

5)☐ Notice of Informal Patent Application (PTO-152)

6)☐ Other: _____.

DETAILED ACTION

1.   Claims 1-6, and 8-15 have been presented for examination.

Claim 7 has been cancelled.

Response to Arguments

2.   A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR
1.17(e), was filed in this application after final rejection. Since this application is eligible for continued
examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the
finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's
submission filed on 22 September 2006 has been entered.

        i)   Applicant's arguments filed 22 September 2006 have been fully considered but they are
not persuasive.

        ii)   Applicant argues that the reference does not the limitations of Claim 1. Applicant's
arguments are not persuasive because they amount to a general allegation that the claims define a
patentable invention without specifically pointing out how the language of the claims patentably
distinguishes them from the references. Applicant's arguments do not clearly point out the patentable
novelty which he or she thinks the claims present in view of the state of the art disclosed by the references
cited or the objections made. Further, they do not show how the amendments avoid such references or
objections.

        The Applicant has merely provided the limitations of Claim 1 and a summary of the reference.
The Applicant then merely states that the reference does not disclose the provided limitations. No
additional arguments are provided to differentiate the claims from the prior art. As stated in the previous
office action, the reference discloses in columns 141 and 142, business logic reuse of certain components
as well as wrapping the components. The reference also discloses utilizing automation in the architecture
for the wrapping process as can be seen in columns 217-218 and figure 83, for example.

iii)   Applicants have not argued the merits of the dependent claims.

Claim Rejections - 35 USC § 101

35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition
of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the
conditions and requirements of this title.

MPEP 2106 recites:

The claimed invention as a whole must accomplish a practical application. That is, it must

produce a "useful, concrete and tangible result" State Street 149 F.3d at 1373, 47 USPQ2d at 1601-02. A

process that consists solely of the manipulation of an abstract idea is not concrete or tangibles. See In re

Warmerdam, 33 F.3d 1354, 1360, 31 USPQ2d 1754, 1759 (Fed.Cir. 1994). See also Schrader, 22 F.3d at

295, 30 USPQ2d at 1459.

3.   **Claims 1-6, and 8-15 are rejected** under 35 U.S.C. 101 because the claimed invention is

directed to non-statutory subject matter.

i)   Claims 1, 6, and 15 recite a method, apparatus, and recording medium which extract,

identify, generate, calculate, determine, an search. As such the claims do not produce a useful concrete,

and tangible result.

ii)   Claim 15 recites a recording medium capable of being read. It is noted that a capability is

not considered a limitation. In addition the claim appears to recite a computer program. It should be noted

that code (i.e., a computer software program) does not do anything per se. Instead, it is the code stored on

a computer that, *when executed*, instructs the computer to perform various functions. The following claim

is a generic example of a proper computer program product claim;

A computer program product embodied on a computer-readable medium and comprising code

that, when executed, causes a computer to perform the following:

> Function A
> Function B
> Function C, etc...

Appropriate correction is required.

All claims dependent upon a rejected base claim are rejected by virtue of their dependency.

Claim Rejections - 35 USC § 112

The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

4.   **Claim 4, 8 and 14 are rejected** under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

   i)   Claim 4 contains the term "etc." therefore the claim is rendered vague and indefinite.

   ii)   Claim 4 contains the phrase "transforms different characters of ASCII, EBCDIC, etc." It is unclear how the characters are being transformed and into what and therefore the claim is rendered vague and indefinite.

   iii)   Claim 14 contains the phrase "Unique Path" and Claim 8 contains the phrase "Top-Down." These phrases appear to be proper nouns. It is unclear if the phrases are attempting to be indicative of a unique definition to these phrases or merely defined as stated.

   iv)   Claim 14 contains the phrase "Critical variable." This phrase appears to be a proper noun. This would therefore raise the possibility that it is differentiable from subsequent phrases, which are not capitalized, and therefore not proper nouns and would raise the issue of lack of antecedent basis.

Appropriate correction is required.

All claims dependent upon a rejected base claim are rejected by virtue of their dependency.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis

for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

5.   **Claims 1-6, 8-10, and 13-15 are rejected** under 35 U.S.C. 102(e) as being clearly anticipated by

Bowman-Amuah "System, Method, and Article of Manufacture for a Globally Addressable

Interface in a Communication Services Patterns Environment", U.S. Patent No. 6,289,382,

hereafter referred to as "Bowman-Amuah".

6.   **Claims 1-6, 8-10, and 13-15 are rejected** under 35 U.S.C. 102(b) as being clearly anticipated by

C. Verhoef "Towards automated modification of legacy assets." hereafter referred to as Verhoef.

7.   **Claims 1-6, 8-10, and 13-15 are rejected** under 35 U.S.C. 102(a) as being clearly anticipated by

"Web Service Workflow" Individual Project Report, June 2001, Dinesh Ganesarajah hereafter

referred to as Dinesh.

Regarding Claim 1:

The reference discloses An apparatus for wrapping existing procedure oriented program into component based system, comprising:

a code analyzing portion for extracting information necessary for program analysis in source program or codes implemented with source procedural language; **(Bowman-Amuah. Column 44, Lines 25-32, and 51-61. Column 50, Lines 60-67. Column 52, Lines 61-67. Column 138, Lines 56-65. Column 141, Lines 17-35. Column 215, Lines 36-46. Figure 66 and 145) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)**

a business logic identifying portion for

calculating the fitting index of user requirements using weighting values of the constituent elements inputted by the user depending on a scale of each module; **(Bowman-Amuah. Column 44, Lines 25-32, and 51-61. Column 50, Lines 60-67. Column 52, Lines 61-67. Column 138, Lines 56-65. Column 141, Lines 17-35. Column 215, Lines 36-46. Figure 66 and 145) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)**

searching the flows within the program for executing a module in the program including the module where the fitting index is the largest; **(Bowman-Amuah. Column 44, Lines 25-32, and 51-61. Column 50, Lines 60-67. Column 52, Lines 61-67. Column 138, Lines 56-65. Column 141, Lines 17-35. Column 215, Lines 36-46. Figure 66 and 145) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)**

searching input/output variables based on variables associated with screen decoration; (Bowman-Amuah. Column 44, Lines 25-32, and 51-61. Column 50, Lines 60-67. Column 52, Lines 61-67. Column 138, Lines 56-65. Column 141, Lines 17-35. Column 215, Lines 36-46. Figure 66 and 145) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)

identifying automatically variables necessary for constraint conditions and interface conditions using input/output variables and flows within the searched program; (Bowman-Amuah. Column 44, Lines 25-32, and 51-61. Column 50, Lines 60-67. Column 52, Lines 61-67. Column 138, Lines 56-65. Column 141, Lines 17-35. Column 215, Lines 36-46. Figure 66 and 145) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)

defining the variables to be constraint condition and variables to be interface condition using the identified variables, and(Bowman-Amuah. Column 44, Lines 25-32, and 51-61. Column 50, Lines 60-67. Column 52, Lines 61-67. Column 138, Lines 56-65. Column 141, Lines 17-35. Column 215, Lines 36-46. Figure 66 and 145) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)

identifying a portion of very high probability of reuse using the information necessary for program analysis extracted in the code analyzing portion; (Bowman-Amuah. Column 44, Lines 25-32, and 51-61. Column 50, Lines 60-67. Column 52, Lines 61-67. Column 138, Lines 56-65. Column 141,

Lines 17-35. Column 215, Lines 36-46. Figure 66 and 145) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2) and

a component wrapper generating portion for automatically generating the codes for wrapping the program workflow which includes business logic identified in the business logic identifying portion. (Bowman-Amuah. Column 44, Lines 25-32, and 51-61. Column 50, Lines 60-67. Column 52, Lines 61-67. Column 138, Lines 56-65. Column 141, Lines 17-35. Column 215, Lines 36-46. Figure 66 and 145) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)

Regarding Claim 2:

The reference discloses The apparatus as claimed in claim 1, wherein the component wrapper generating portion comprises:

a component framework for reusing existing system as a component; (Bowman-Amuah. Column 215, Lines 36-46) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)

a legacy frame work which is a framework of system to be associated with the component framework; (Bowman-Amuah. Column 215, Lines 36-46. Column 138, Lines 56-65) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section

3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2) and

an intermediate framework for linking the component framework with the legacy frame work, and capturing screen information which is input/output to/from the legacy framework, thereby automatically inserting or extracting information. **(Bowman-Amuah. Column 215, Lines 36-46. Column 138, Lines 56-65) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)**

Regarding Claim 3:

The reference discloses The apparatus as claimed in claim 2, wherein the intermediate framework comprises:

a program scheduler having navigation information and interaction relationship between programs, and having schedule information about whether a plurality of screens are for input or for output; **(Bowman-Amuah. Column 50, Lines 60-67) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)**

a meta-data pool for storing meta-information for the screens of programs included in a pre-registered workflow; **(Bowman-Amuah. Column 41, Lines 25-31. Figure 66) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)**

a record handler for analyzing the command required by the component framework, obtaining

the meta information of input/output data from the meta-data pool, thereby finding which are the screens

entered from the present existing system and which are the input/output data corresponding to the screens,

and for transferring the input/output data; and **(Bowman-Amuah. Column 52, Lines 16-21. Figure 145)**

(Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page

31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-

61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106,

Section 9.4.2)

a record adapter for receiving input screen from the legacy component, differentiating the data

associated with the input/output from the information for display of screen only, and providing it to the

record handler. **(Bowman-Amuah. Column 44, Lines 51-61) (Verhoef. Page 315, Introduction. Page**

325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section

4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2.

Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)

Regarding Claim 4:

The reference discloses The apparatus as claimed in claim 3, wherein the record adapter stores

temporarily the information of legacy component, and transforms different characters of ASCII,

EBCDIC,etc.

(Bowman-Amuah. Column 56, Lines 12-18) (Verhoef. Page 315, Introduction. Page 325, Section 4.4.

Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43,

Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87,

Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)

Regarding Claim 5:

The reference discloses The apparatus as claimed in claim 3, wherein the program scheduler stores the information about how the program is used for any usage of input, output or input/output usage. (Bowman-Amuah. Column 50, Lines 60-67) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)

Regarding Claim 6:

The reference discloses A method for wrapping existing procedure oriented program into component based system, comprising the steps of:

extracting information necessary for program analysis in source program or codes implemented with source procedural language; (Bowman-Amuah. Column 20, Lines 20-28. Column 44, Lines 25-32, and 51-61. Column 50, Lines 60-67. Column 52, Lines 61-67. Column 138, Lines 56-65. Column 141, Lines 17-35. Column 215, Lines 36-46. Figure 66 and 145) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)

identifying a portion of very high probability of reuse using the information necessary for program analysis extracted in the code analyzing portion; (Bowman-Amuah. Column 20, Lines 20-28. Column 44, Lines 25-32, and 51-61. Column 50, Lines 60-67. Column 52, Lines 61-67. Column 138, Lines 56-65. Column 141, Lines 17-35. Column 215, Lines 36-46. Figure 66 and 145) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section

3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2) and

generating automatically the codes for wrapping program workflow which include business logic identified in the business logic identifying portion. **(Bowman-Amuah. Column 20, Lines 20-28. Column 44, Lines 25-32, and 51-61. Column 50, Lines 60-67. Column 52, Lines 61-67. Column 138, Lines 56-65. Column 141, Lines 17-35. Column 215, Lines 36-46. Figure 66 and 145) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)**

wherein the step of identifying comprises the steps of:

calculating the fitting index of user requirement using weighting value of the constituent elements inputted by user depending on a scale of each module in order to express business type to be identified; **(Bowman-Amuah. Column 20, Lines 20-28. Column 44, Lines 25-32, and 51-61. Column 50, Lines 60-67. Column 52, Lines 61-67. Column 138, Lines 56-65. Column 141, Lines 17-35. Column 215, Lines 36-46. Figure 66 and 145) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)**

determining where the calculated fitting index is the largest, then searching the flows within program for executing module in the program including the module where the fitting index is the largest, **(Bowman-Amuah. Column 20, Lines 20-28. Column 44, Lines 25-32, and 51-61. Column 50, Lines 60-67. Column 52, Lines 61-67. Column 138, Lines 56-65. Column 141, Lines 17-35. Column 215, Lines 36-46. Figure 66 and 145) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2.**

Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page

100, Section 8.3.3. Page 106, Section 9.4.2)

searching input/output variables based on variables associated with screen decoration having

the direct relations with user; (Bowman-Amuah. Column 20, Lines 20-28. Column 44, Lines 25-32,

and 51-61. Column 50, Lines 60-67. Column 52, Lines 61-67. Column 138, Lines 56-65. Column 141,

Lines 17-35. Column 215, Lines 36-46. Figure 66 and 145) (Verhoef. Page 315, Introduction. Page

325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section

4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2.

Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)

identifying automatically variables necessary for constraint condition and interface using

input/output variables and flows (path) within the searched program; and(Bowman-Amuah. Column 20,

Lines 20-28. Column 44, Lines 25-32, and 51-61. Column 50, Lines 60-67. Column 52, Lines 61-67.

Column 138, Lines 56-65. Column 141, Lines 17-35. Column 215, Lines 36-46. Figure 66 and 145)

(Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page

31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-

61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106,

Section 9.4.2)

defining the variables to be constraint condition and variables to be interface using the identified

variables, to generate the code for the wrapping. (Bowman-Amuah. Column 20, Lines 20-28. Column

44, Lines 25-32, and 51-61. Column 50, Lines 60-67. Column 52, Lines 61-67. Column 138, Lines 56-

65. Column 141, Lines 17-35. Column 215, Lines 36-46. Figure 66 and 145) (Verhoef. Page 315,

Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4.

Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page

76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)

Regarding Claim 8:

The reference discloses The method as claimed in claim 6, wherein the calculation of the fitting index in the step of calculating the fitting index, lies in that it calculates fitness (fitting index) about the user requirement in Top-Down method which searches form large portion to small portion in scale. (Bowman-Amuah. Column 163, Lines 51-62) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)

Regarding Claim 9:

The reference discloses The method as claimed in claim g 6, wherein the constraint condition consists of control variables necessary to obtain the flow for executing the module of the desired business logic, and wherein the interface consists of variables utilized in input/output portion of data.

(Bowman-Amuah. Column 141, Lines 22-35) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)

Regarding Claim 10:

The reference discloses The method as claimed in claim 6, wherein in the step of searching the flows within the program and the input/output variables, the step of searching the flows within the program comprises the steps of:

collecting the flow information between paragraphs and the information about the condition

thereof by the unit of modules and paragraphs in order to adjust the inferior information of the program

systematically when the programs for the detailed analysis of program, the least modules having business

logics, and the paragraph candidates are known, searching call relations between the paragraphs using

function call statement for searching call relations between the modules; **(Bowman-Amuah. Column 68,**

Lines 30-45) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3)

(Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section

5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3.

Page 106, Section 9.4.2)

eliminating redundancy or recursive portions of the paragraph calls for inclusive call relations,

then reconstructing the paragraph call relations; **(Bowman-Amuah. Column 74, Lines 35-38) (Verhoef.**

Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32,

Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61.

Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section

9.4.2)

identifying the flow of the program taking account of only unstructured statement sentence of the

flow information between the paragraphs for searching the paragraph flow by the unstructured statement;

(Bowman-Amuah. Column 74, Lines 35-38) (Verhoef. Page 315, Introduction. Page 325, Section 4.4.

Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43,

Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87,

Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2) and

generating call relation tree using the call relation information of the paragraph acquired and the

program flow information of the unstructured sentence. **(Bowman-Amuah. Column 266, Lines 1-9)**

(Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page

31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-

61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106,

Section 9.4.2)

Regarding Claim 13:

The reference discloses The method as claimed in claim 6, wherein in the step of searching the

flows within program and the input/output variables, the step of searching the input/output variables

comprises the steps of:

analyzing the screen information of each variable and field, by analyzing the input/output

variables which exist in the program having business logic to be reused, the information about user

interface or forms for expressing a screen; **(Bowman-Amuah. Column 38, Lines 17-28. Figure 75)**

(Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page

31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-

61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106,

Section 9.4.2)

determining whether or not a field exists in the analyzed screen information; **(Bowman-Amuah.**

Column 38, Lines 17-28. Figure 75) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page

327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section

5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57.

Page 100, Section 8.3.3. Page 106, Section 9.4.2)

discriminating whether the field is a portion for input/output of actual data or only for decoration

of screen when the field exists in the analyzed screen information; and **(Bowman-Amuah. Column 38,**

Lines 17-28. Figure 75) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238,

Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page

59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100,

Section 8.3.3. Page 106, Section 9.4.2)

 a) registering, if the field is for input/output (I/O) field, the field as an input/output variable since

the field is used as input/output variable, **(Bowman-Amuah. Column 38, Lines 17-28. Figure 75)**

(Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page

31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-

61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106,

Section 9.4.2)

 b) registering, if the field is not for the portion for input/output, the field as meta data since the

field is used as decoration of the screen. **(Bowman-Amuah. Column 38, Lines 17-28. Figure 75)**

(Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page

31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-

61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106,

Section 9.4.2)

Regarding Claim 14:

 The reference discloses The method as claimed in claim 6, wherein the step of identifying

automatically variables necessary for constraint condition and interface, comprises the steps of:

 selecting Unique Path having the workflow that user wants in the generated tree; **(Bowman-**

Amuah. Column 169, Lines 59-67. Column 170, Lines 1-4) (Verhoef. Page 315, Introduction. Page

325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section

4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2.

Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)

checking whether there exist Critical variable that decide the paragraph flow or input/output in the designated workflow; **(Bowman-Amuah. Column 185, Lines 61-67) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)**

tracking, for the critical variable, the list of variables affecting the critical variable using impact analysis, or tracking, for the variables transferred between programs, continuously the calling programs or the called programs and identifying the usage of variables; **(Bowman-Amuah. Column 63, Lines 28-37. Figure 149) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)**

discriminating whether the identified variables are those of determining workflow path or those utilized as a constraint condition; and

a) for the identified variables used as a control variable, adding them to the list of the control variables, **(Bowman-Amuah. Column 120, Lines 10-18. Column 121, Lines 52-63) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)**

b) for the identified variables used as a constraint condition, adding them to the list of the a constraint condition. **(Bowman-Amuah. Column 120, Lines 10-18. Column 121, Lines 52-63) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-**

61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106,

Section 9.4.2)

Regarding Claim 15:

The **reference discloses** A recording medium capable of being read by a digital processing

apparatus, in which programs capable of being executed by the digital processing apparatus are

implemented by types so as to perform a method for wrapping existing procedure oriented program into

component based system, wherein the method comprises the steps of:

extracting information necessary for program analysis in source program or codes

implemented with source procedural language; **(Bowman-Amuah. Column 20, Lines 20-28. Column**

44, Lines 25-32, and 51-61. Column 50, Lines 60-67. Column 52, Lines 61-67. Column 138, Lines 56-

65. Column 141, Lines 17-35. Column 215, Lines 36-46. Figure 66 and 145) (Verhoef. Page 315,

Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4.

Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page

76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)

identifying a portion of very high probability of reuse using the information necessary for

program analysis extracted in the code analyzing portion; **(Bowman-Amuah. Column 20, Lines 20-28.**

Column 44, Lines 25-32, and 51-61. Column 50, Lines 60-67. Column 52, Lines 61-67. Column 138,

Lines 56-65. Column 141, Lines 17-35. Column 215, Lines 36-46. Figure 66 and 145) (Verhoef. Page

315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section

3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4.

Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2) and

generating automatically the codes for wrapping program workflow which includes

business logic identified in the business logic identifying portion, **(Bowman-Amuah. Column 20, Lines**

20-28. Column 44, Lines 25-32, and 51-61. Column 50, Lines 60-67. Column 52, Lines 61-67.

Column 138, Lines 56-65. Column 141, Lines 17-35. Column 215, Lines 36-46. Figure 66 and 145)

(Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page

31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-

61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106,

Section 9.4.2)

the step of identifying comprising the steps of.

calculating the fitting index of user requirement using weighting value of the constituent elements

inputted by user depending on a scale of each module in order to express business type to be identified;

(Bowman-Amuah. Column 20, Lines 20-28. Column 44, Lines 25-32, and 51-61. Column 50, Lines

60-67. Column 52, Lines 61-67. Column 138, Lines 56-65. Column 141, Lines 17-35. Column 215,

Lines 36-46. Figure 66 and 145) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-

238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2.

Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page

100, Section 8.3.3. Page 106, Section 9.4.2)

determining whether the calculated fitting index is the largest then searching, for the largest

fitting index, the flows within program for executing module in the program including the module where

the fitting index is the largest, (Bowman-Amuah. Column 20, Lines 20-28. Column 44, Lines 25-32,

and 51-61. Column 50, Lines 60-67. Column 52, Lines 61-67. Column 138, Lines 56-65. Column 141,

Lines 17-35. Column 215, Lines 36-46. Figure 66 and 145) (Verhoef. Page 315, Introduction. Page

325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section

4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2.

Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)

searching input/output variables based on variables associated with screen decoration having the direct relations with user; (Bowman-Amuah. Column 20, Lines 20-28. Column 44, Lines 25-32, and 51-61. Column 50, Lines 60-67. Column 52, Lines 61-67. Column 138, Lines 56-65. Column 141, Lines 17-35. Column 215, Lines 36-46. Figure 66 and 145) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)

identifying automatically variables necessary for constraint condition and interface using input/output variables and flows (path) within the searched program; and(Bowman-Amuah. Column 20, Lines 20-28. Column 44, Lines 25-32, and 51-61. Column 50, Lines 60-67. Column 52, Lines 61-67. Column 138, Lines 56-65. Column 141, Lines 17-35. Column 215, Lines 36-46. Figure 66 and 145) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)

defining the variables to be constraint condition and variables to be interface using the identified variables, to generate the code for the wrapping. (Bowman-Amuah. Column 20, Lines 20-28. Column 44, Lines 25-32, and 51-61. Column 50, Lines 60-67. Column 52, Lines 61-67. Column 138, Lines 56-65. Column 141, Lines 17-35. Column 215, Lines 36-46. Figure 66 and 145) (Verhoef. Page 315, Introduction. Page 325, Section 4.4. Page 327-238, Section 5.3) (Dinesh. Page 31-32, Section 3.6.4. Page 38, Section 4.3.2. Page 43, Section 5.2.2. Page 59, Section 5.7.3. Page 60-61. Section, 5.7.4. Page 76, Section 7.2.2. Page 87, Figure 57. Page 100, Section 8.3.3. Page 106, Section 9.4.2)

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness

rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as
> set forth in section 102 of this title, if the differences between the subject matter sought to be
> patented and the prior art are such that the subject matter as a whole would have been obvious at
> the time the invention was made to a person having ordinary skill in the art to which said subject
> matter pertains. Patentability shall not be negatived by the manner in which the invention was
> made.

The factual inquiries set forth in *Graham* v. *John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966),

that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are

summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

This application currently names joint inventors. In considering patentability of the claims under
35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly
owned at the time any inventions covered therein were made absent any evidence to the contrary.
Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of
each claim that was not commonly owned at the time a later invention was made in order for the examiner
to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under
35 U.S.C. 103(a).

8. **Claim(s) 11 and 12** are rejected under 35 U.S.C. 103(a) as being unpatentable over **Bowman-**

Amuah "System, Method, and Article of Manufacture for a Globally Addressable Interface in a

Communication Services Patterns Environment", U.S. Patent No. 6,289,382, hereafter referred to

as "Bowman-Amuah".

9. **Claim(s) 11 and 12** are rejected under 35 U.S.C. 103(a) as being unpatentable over **C. Verhoef**

"Towards automated modification of legacy assets." hereafter referred to as Verhoef.

Regarding Claim 11:

The references do not explicitly disclose The method as claimed in claim 7, wherein in the step of searching call relations between the paragraphs, the function call statement utilizes CALL sentence and PERFORM sentence in COBOL.

However, it would have been obvious to one of ordinary skill in the art at the time of the invention to utilize function call statements that are present in COBOL, which is disclosed in the prior art in order to allow for determining paragraph call relations. **(Bowman-Amuah. Column 115, Line 6) (Verhoef. Page 318, Section 1.3)**

Regarding Claim 12:

The references do not explicitly disclose The method as claimed in claim 10, wherein in the step of identifying the flow of the program, the unstructured statement utilizes at least one of GO TO sentence, CONTINUE sentence and BREAK sentence in COBOL.

However, it would have been obvious to one of ordinary skill in the art at the time of the invention to utilize unstructured statements that are present in COBOL, which is disclosed in the prior art in order to allow for identification of the flow of the program. **(Bowman-Amuah. Column 115, Line 6) (Verhoef. Page 318, Section 1.3)**

10.   Claim(s) 11 and 12 are rejected under 35 U.S.C. 103(a) as being unpatentable over "Web Service Workflow" Individual Project Report, June 2001, Dinesh Ganesarajah hereafter referred to as Dinesh.

Regarding Claim 11:

The reference does not explicitly disclose The method as claimed in claim 7, wherein in the step of searching call relations between the paragraphs, the function call statement utilizes CALL sentence and PERFORM sentence in COBOL.

However, it would have been obvious to one of ordinary skill in the art at the time of the invention to utilize function call statements that are present in COBOL, which is a popular business oriented programming language in order to allow for determining paragraph call relations.

Regarding Claim 12:

The reference does not explicitly disclose The method as claimed in claim 10, wherein in the step of identifying the flow of the program, the unstructured statement utilizes at least one of GO TO sentence, CONTINUE sentence and BREAK sentence in COBOL.

However, it would have been obvious to one of ordinary skill in the art at the time of the invention to utilize unstructured statements that are present in COBOL, which is a popular business oriented programming language in order to allow for identification of the flow of the program.

Conclusion

11.   All Claims are rejected.

12.   Any inquiry concerning this communication or earlier communications from the examiner should be directed to Saif A. Alhija whose telephone number is (571) 272-8635. The examiner can normally be reached on M-F, 11:00-7:30.
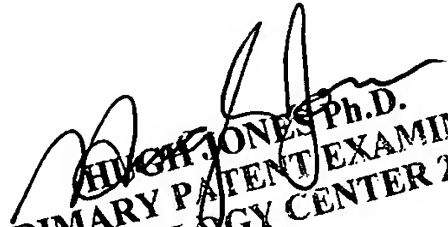
If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kamini Shah can be reached on (571) 272-22792279. The fax phone number for the organization where this application or proceeding is assigned is (571) 273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

SAA

November 6, 2006